

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

### **REMARKS/ARGUMENTS**

In the Office Action, the Examiner noted that claims 1-20 are pending in the application. The Examiner additionally stated that claims 1-20 are rejected. By this amendment, claims 1, 6, 9, 11-12, 15, and 18-19 have been amended. Hence, claims 1-20 are pending in the application.

Applicant hereby requests further examination and reconsideration of the application, in view of the foregoing amendments.

#### **In the Claims**

##### **Rejections Under 35 U.S.C. §112**

The Examiner rejected claims 1-20 under 35 U. S.C. 1 12, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Regarding claims 1, 6, and 11, the Examiner noted that Applicant has not ruled out that  $P$  cannot equal 0, and furthermore stated that if  $P = 0$ , then the claim is unclear as it is not clear how the system could fetch from 0 streams. In addition, the Examiner remarked that if Applicant did intend to include  $P = 0$ , then the claim also has enablement issues. The Examiner moreover noted with regard to claims 1 and 6, that it is not clear how a fetch algorithm can be coupled to predictors (hardware), giving example that an algorithm is merely a set of steps which is to be performed and that a set of steps cannot be coupled to hardware.

In response, Applicant has amended independent claims 1, 6, and 11 to recite that instructions are fetched from one to  $P$  of the multiple hardware streams to a pipeline. In addition, Applicant has amended independent claims 1 and 6 to recite that it is a fetch stage (hardware) that fetches instructions into the pipeline.

In view of the above noted amendments to independent claims 1, 6, and 11, Applicant respectfully requests that the rejections of claims 1-20 be withdrawn.

##### **Rejections Under 35 U.S.C. §102(b)**

The Examiner rejected claims 1-14 and 18-19 under 35 U.S.C. §102(b) as being anticipated by Yoaz et al., "Speculative Techniques for Improving Load Related

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

Scheduling,” May 1999 (as applied in the previous Office action and herein referred to as Yoaz). In addition, the Examiner cited Parady, U.S. Patent 5,933,627 (as applied in the previous Office Action and hereinafter referred to as “Parady”) as extrinsic evidence for showing that it is common to have separate hardware streams for each thread. Applicant respectfully traverses the Examiner’s rejections.

Prior to providing a claim-by-claim analysis, a brief summary of the teachings of Yoaz and Parady are provided below, vis-à-vis the invention disclosed by Applicant in the instant application. This information is provided to aid the Examiner during reconsideration of the claims.

Yoaz discloses three techniques to address *instruction scheduler* limitations in an out-of-order engine, where the *instruction scheduler* is responsible for *dispatching instructions to execution units* based upon dependencies, latencies, and resource availability. The problem that is noted by Yoaz, and which motivates his *instruction scheduler* techniques, is that dynamic latencies of load instructions are unknown, so *scheduling* dependent instructions is based on either load-use delay or pessimistic delay. (cf. Abstract) Yoaz expands his teachings in the area of hit/miss predictions by noting that the new concept presented is to predict which loads will miss the cache, thus *delaying the dependent instructions* until the needed data is fetched. Yoaz opines that this increases performance directly by saving a few clocks through *the scheduling of load-dependent instructions to execute* at the exact time the data is retrieved. Yoaz’s technique involves a hardware approach that is based on *a per-load binary prediction* of a hit or miss in the cache. (cf. page 44, lines 8-27) The author specifically notes that once all instruction dependencies have been resolved, the *scheduler’s* remaining task is to *dispatch instructions* in a way that maximizes the utilization of available resources, while minimizing instruction latencies. Further, Yoaz suggests dynamically predicting whether a specific load will hit or miss the cache, *to facilitate the scheduling operation*. (cf. page 46, lines 17-31) Yoaz moreover intimates that multi-threading may benefit from hit/miss prediction, and that the prediction may be used to govern a thread switch if a load is predicted to miss the L2 cache. (cf. page 47, lines 25-29)

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

Parady discloses a method and apparatus for switching between threads of a program *in response to a long-latency event*. In one embodiment, the long-latency events are load or store operations which trigger a thread switch *if there is a miss* in the level 2 cache. In addition to providing separate groups of registers for multiple threads, a group of program address registers pointing to different threads are provided. A switching mechanism switches between the program address registers *in response to* the long-latency events. (cf. Abstract) Parady also defines the process whereby a multithreading processor interleaves threads in such a manner as described above (i.e., in response to a long-latency event) as “coarse-grain multithreading,” (cf. col. 2, lines 8-10) and furthermore teaches the concept of “a switching mechanism [that] switches between the program address registers *in response to* the long-latency events. Parady discloses the switching mechanism as “[t]hread switching logic 112 provided to give a hardware thread switching capability. The indication that a thread switch is required is provided on a line 114 providing an *L2-miss indication from cache control/system interface 22*.” He further teaches that “[u]pon such an indication, a switch to the next thread will be performed.” (cf. col. 3, lines 57-62) Furthermore, in a background discussion of his invention, Parady cites an IBM article that distinguishes between processors to which his invention is directed (i.e., coarse-grain multithreaded processors) and fine-grain multithreaded processors, that is, those processors which interleave threads on a cycle-by-cycle basis. (cf. col. 2, lines 6-10) Parady’s invention and disclosure is directed towards problems associated with coarse-grain processors: those processor that switch threads *in response to* long latency events.

In contrast to the teachings of Yoaz and Parady, Applicant’s invention is directed towards a processor having multiple hardware streams supporting multiple data threads, and a data cache. In this processor, Applicant discloses a system for *fetching* instructions from one to P of the multiple hardware streams to a pipeline, where P is less than the number of multiple hardware streams. The system has multiple hit/miss predictors and a fetch stage. The multiple hit/miss predictors are each associated with a corresponding one of the multiple hardware streams, and are each configured to forecast whether corresponding instructions from the corresponding one of the multiple hardware streams

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

will hit or miss the data cache. The multiple hit/miss predictors forecast whether the corresponding instructions from the corresponding one of the multiple hardware streams will hit or miss the data cache prior to when the corresponding instructions enter into a dispatch stage in the pipeline. The fetch stage is coupled to the multiple hit/miss predictors. The fetch stage simultaneously fetches every cycle, the instructions from the one to P of the multiple hardware streams to the pipeline and furthermore selects, on a cycle-by-cycle basis, the one to P of the multiple hardware streams from which to fetch the instructions.

Yoaz's technique involving hit/miss prediction is clearly directed towards scheduling instructions for execution which have already been fetched into the pipeline and that have entered a dispatch stage for scheduling for execution. Such a technique is disadvantageous because the instructions have already entered the pipeline and have reached the dispatch stage. If Yoaz's hit/miss prediction results in a miss prediction, then instructions between the fetch stage and the dispatch stage must be flushed in addition to those within the fetch stage. Yoaz fails even to note the problem that is addressed by Applicant in the instant application, to wit that if the fact that an instruction will miss the data cache could be known early in the process, then the fetching of instructions that might eventually be flushed may be avoided. While it is clear that a hit/miss prediction in a dispatch stage must certainly be coupled to fetch logic in order to redirect the fetching of instructions in the event of a miss prediction, because Yoaz's technique performs the hit/miss predication at dispatch, the pipeline stages above dispatch must be flushed in the event of a miss prediction. Applicant's invention, in contrast, makes the prediction at the fetch stage itself.

It therefore does not follow that Yoaz anticipates a system for making a hit/miss prediction at a fetch stage of a multi-threaded processor pipeline, for his article is directed towards making such predictions at a dispatch stage.

In view of the above summarizations, a claim-by-claim analysis will now be presented.

Amended claim 1 is provided below for ease of reference.

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

1. In a processor having multiple hardware streams supporting multiple data threads, and a data cache, a system for fetching instructions from one to P of the multiple hardware streams to a pipeline, where P is less than the number of multiple hardware streams, the system comprising:

multiple hit/miss predictors, each associated with a corresponding one of the multiple hardware streams, said each configured to forecast whether corresponding instructions from said corresponding one of the multiple hardware streams will hit or miss the data cache, wherein said multiple hit/miss predictors forecast whether said corresponding instructions from said corresponding one of the multiple hardware streams will hit or miss the data cache prior to when said corresponding instructions enter into a dispatch stage in the pipeline; and

a fetch stage, coupled to said multiple hit/miss predictors, configured to simultaneously fetch every cycle, the instructions from the one to P of the multiple hardware streams to the pipeline, and configured to select, on a cycle-by-cycle basis, the one to P of the multiple hardware streams from which to fetch the instructions.

Claim 1 recites, in combination, within a processor having multiple hardware streams supporting multiple data threads, and a data cache, a system for fetching instructions from one to P of the multiple hardware streams to a pipeline. The system has multiple hit/miss predictors that are each associated with a corresponding one of the multiple hardware streams. In addition, each of the multiple hit/miss predictors is configured to forecast whether corresponding instructions from the corresponding one of the multiple hardware streams will hit or miss the data cache. The multiple hit/miss predictors forecast whether said corresponding instructions from said corresponding one of the multiple hardware streams will hit or miss the data cache prior to when said corresponding instructions enter into a dispatch stage in the pipeline. The system additionally includes a fetch stage that is coupled to the multiple hit/miss predictors. The fetch stage simultaneously fetches every cycle, the instructions from the one to P of the

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

multiple hardware streams to the pipeline, and selects, on a cycle-by-cycle basis, the one to P of the multiple hardware streams from which to fetch the instructions.

In response to arguments provided by Applicant filed on 2/22/2005, the Examiner notes that Applicant argues the novelty/rejection of claim 1 on page 13 of the remarks, in substance that:

"It therefore does not follow that Yoaz anticipates a system for fetching instructions into a multi-threaded processor pipeline, for his article is directed towards execution scheduler limitations. Furthermore, to suggest that the teachings of Parady and Yoaz can be combined in a manner relative to Applicant's invention is out of place because Yoaz teaches how to schedule instructions for execution and Parady's invention is directed towards responding to the occurrence of long-latency events." These arguments are not found persuasive for the following reasons: a) Just because Yoaz discusses scheduling does not mean that Yoaz does not also teach applicant's claimed invention. More specifically, scheduling and fetching are directly related. If a certain thread is to be scheduled, then instructions from that thread must be fetched. Yoaz is concerned with switching threads on a cache miss prediction and fetching/scheduling instructions from a next thread. Also, that examiner has used Parady to show nothing more than the fact that threads may be assigned to different streams. With respect to the rejections, Parady's response to long-latency events does not come into play.

Applicant responds that Yoaz's teaching is restricted to making hit/miss predictions when instructions enter the dispatch stage of a pipeline where the instructions, having already been fetched into the pipeline and provided to the dispatch stage, are scheduled for execution.. Yoaz does not address making such predictions higher up in the pipeline, at the fetch stage, to circumvent the problems associated with flushing stages between fetch and dispatch in the event of a miss prediction.

For these reasons, Applicant respectfully requests that the Examiner withdraw his rejection of claim 1.

With respect to claims 2-5 and 14, these claims depend from claim 1 and add further limitations that are neither anticipated nor made obvious by Yoaz, Parady, or Parady and

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

Yoaz in combination. Accordingly, Applicant respectfully requests that the Examiner withdraw his rejections to claims 2-5 and 14.

In a manner substantially similar to claim 1, claim 6 recites, in combination with other elements, a data cache, comprising a plurality of levels; multiple hit/miss predictors, and a fetch stage. The multiple hit/miss predictors are each associated with a corresponding one of the multiple hardware streams, and each are configured to forecast whether corresponding instructions from the corresponding one of the multiple hardware streams will hit or miss said data cache. The said multiple hit/miss predictors forecast whether the corresponding instructions from the corresponding one of the multiple hardware streams will hit or miss said data cache prior to when said corresponding instructions enter into a dispatch stage in a pipeline of the processor. These elements and limitations are entirely absent from the teachings of Yoaz because Yoaz's technique for hit/miss prediction is performed at the dispatch level on instructions that have already been provided by fetch logic to the pipeline. Consequently, for reasons substantially noted above in arguments presented in traversal of the Examiner's rejection of claim 1, Applicant asserts with respect to the rejection of claim 6 that Yoaz teaches a technique to improve the scheduling of instructions for execution where hit/miss prediction is performed at the dispatch stage. Applicant's invention, on the other hand, is directed towards improved instruction fetching in a multithreaded processor, prior to providing instructions to the pipeline for dispatch.

For these reasons, Applicant respectfully requests that the Examiner withdraw his rejection of claim 6.

With respect to claims 7-10, these claims depend from claim 6 and add further limitations that are neither anticipated nor made obvious by Yoaz, Parady, or Yoaz and Parady in combination. Accordingly, Applicant respectfully requests that the Examiner withdraw his rejections to claims 7-10.

Like claims 1 and 6, claim 11 recites, a method for simultaneously fetching instructions every cycle from up to P hardware streams to a pipeline. The method includes, for each of the hardware streams, making a hit/miss prediction by a corresponding one of

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

associated hit/miss predictors as to whether corresponding instructions for the each of the multiple hardware streams previously fetched will hit or miss the data cache, where the making of the hit/miss prediction is performed prior to when the corresponding instructions enter into a dispatch stage in the pipeline; and selecting, on a cycle-by-cycle basis, the P hardware streams from which to fetch the instructions. Neither Yoaz nor Parady suggests anything about these above noted elements, in particular that hit/miss predictions are made prior to when instructions are provided by the pipeline for dispatch. Consequently, for reasons substantially noted above in arguments presented in traversal of the Examiner's rejections of claim 1 and claim 6, Applicant asserts with respect to the rejection of claim 11 that Yoaz teaches how to more effectively schedule instructions *that have already been fetched into a pipeline*, for execution. Applicants invention makes hit/miss predictions prior to when instructions enter the dispatch stage.

Accordingly, Applicant respectfully requests that the Examiner withdraw his rejection of claim 11.

With respect to claims 12-13 and 18-19, these claims depend from claim 11 and add further limitations that are neither anticipated nor made obvious by Yoaz, Parady, or Yoaz and Parady in combination. Accordingly, Applicant respectfully requests that the Examiner withdraw his rejections to claims 12-13 and 18-19.

#### **Rejections Under 35 U.S.C. §103(a)**

The Examiner rejected claim 15 under 35 U.S.C. §103(a) as being unpatentable over Yoaz, as applied in the rejections under 35 U.S.C. §102 discussed above. Applicant respectfully traverses and notes that Yoaz does not teach the specific limitation recited in claim 1 that the multiple hit/miss predictors forecast whether said corresponding instructions from said corresponding one of the multiple hardware streams will hit or miss the data cache prior to when said corresponding instructions enter into a dispatch stage in the pipeline. Accordingly, since claim 15 adds further limitations over that recited in claim 1, it is respectfully requested that the rejection of claim 15 be withdrawn.

The Examiner also rejected claims 16-17 and 20 under 35 U.S.C. §103(a) as being unpatentable over Yoaz, as applied in the rejections under 35 U.S.C. §102 discussed



Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

above, in view of Ryan, U.S. Patent No. 5,694,572. Applicant respectfully traverses and notes that Yoaz does not teach the limitations and elements recited in claims 1, 6, or 11, as noted in arguments provided above. Accordingly, since claim 16 adds further limitations over that recited in claim 1, it is respectfully requested that the rejection of claim 16 be withdrawn. Likewise, since claim 17 adds further limitations over that recited in claim 6, it is respectfully requested that the rejection of claim 17 be withdrawn. In addition, since claim 20 adds further limitations over that recited in claim 11, it is respectfully requested that the rejection of claim 20 be withdrawn.

Application No. 09/595776 (Docket: MIPS.0166-00-US)  
37 CFR 1.111 Amendment dated 09/30/2005  
Reply to Office Action of 05/13/2005

### **CONCLUSIONS**

In view of the arguments advanced above, Applicant respectfully submits that claims 1-20 are in condition for allowance. Reconsideration of the rejections is requested, and allowance of the claims is solicited.

Applicant earnestly requests that the Examiner contact the undersigned practitioner by telephone if the Examiner has any questions or suggestions concerning this amendment, the application, or allowance of any claims thereof.

I hereby certify under 37 CFR 1.8 that this correspondence is being facsimile transmitted to the United States Patent and Trademark Office on the date of signature shown below.
--

Respectfully submitted,  
**HUFFMAN PATENT GROUP, LLC**

*/ Richard K. Huffman /*

By: \_\_\_\_\_

**RICHARD K. HUFFMAN, P.E.**  
Registration No. 41,082  
Tel: (719) 575-9998

*09/30/2005*

Date: \_\_\_\_\_